SO302 - Package and Boot Environment Administration

Solaris 10 pkgXYZ commands

Introduction

Under Solaris 10, package administration was carried out using the following commands:

Command	Description
pkginfo	Displays information about a package or packages
pkgadd	Installs a package
pkgchk	Checks package installation accuracy
pkgrm	Uninstalls a package

These commands are still available under Solaris 11.

The Solaris 11 Image Packaging System (IPS)

Introduction

Although still present as shown above, use of the Solaris 10 package management commands has been replaced in Solaris 11 by the **Solaris Image Packaging System**.

Solaris 11 uses the **new Image Packaging System** technology combined with the Service Management Facility (SMF), the Zettabyte File System (ZFS) technologies to greatly simplify **both** package management **and** patch management.

The **IPS pkg** command and its subcommands work much like their Linux counterparts **yum**, **apt-get** and **zypper**, relying on a **repository** which contains available packages.

pkg Subcommands

The pkg subcommands are as follows:

Subcommand	Description
publisher	Lists current and disabled repositories “published” by **publishers**
set-publisher	Adds a publisher
unset-publisher	Removes a publisher
list	Lists installed packages
search	Searches for a package in the system's active publishers and returns the first match found
install	Installs/Updates packages
uninstall	Uninstalls packages
refresh	Updates the cached list of available packages
update	Updates all or certain packages to their latest versions
info	Displays information about packages
contents	Displays files installed by a package
verify	Validates the installation and digital signature packages
history	Displays the pkg history
help	Lists the above and advanced subcommands and their switches

LAB #1 - Using IPS

Basic pkg subcommands

Firstly, check which publishers are installed on the system:

```
root@solaris:~# pkg publisher
PUBLISHER                    TYPE     STATUS P LOCATION
solaris                      origin   online F http://pkg.oracle.com/solaris/release/
```

You can display additional information about a publisher as follows:

```
root@solaris:~# pkg publisher solaris

            Publisher: solaris
                Alias:
           Origin URI: http://pkg.oracle.com/solaris/release/
              SSL Key: None
             SSL Cert: None
           Client UUID: ddee2130-0292-11e2-b9e5-80144f013e20
      Catalog Updated: October 16, 2012 09:25:04 PM
              Enabled: Yes
```

Now add the **OpenIndiana** publisher:

```
root@solaris:~# pkg set-publisher -g http://pkg.openindiana.org/sfe sfe
root@solaris:~# pkg publisher
PUBLISHER                   TYPE     STATUS P LOCATION
solaris                     origin   online F http://pkg.oracle.com/solaris/release/
sfe                         origin   online F http://pkg.openindiana.org/sfe/
```

To understand the use of the **-g** switch, use the built-in help function:

```
root@solaris:~# pkg help set-publisher
Usage:
        pkg set-publisher [-Ped] [-k ssl_key] [-c ssl_cert]
            [-g origin_to_add|--add-origin=origin_to_add ...]
            [-G origin_to_remove|--remove-origin=origin_to_remove ...]
            [-m mirror_to_add|--add-mirror=mirror_to_add ...]
            [-M mirror_to_remove|--remove-mirror=mirror_to_remove ...]
            [-p repo_uri] [--enable] [--disable] [--no-refresh]
            [--reset-uuid] [--non-sticky] [--sticky]
            [--search-after=publisher]
            [--search-before=publisher]
            [--search-first]
            [--approve-ca-cert=path_to_CA]
```

```
            [--revoke-ca-cert=hash_of_CA_to_revoke]
            [--unset-ca-cert=hash_of_CA_to_unset]
            [--set-property name_of_property=value]
            [--add-property-value name_of_property=value_to_add]
            [--remove-property-value name_of_property=value_to_remove]
            [--unset-property name_of_property_to_delete]
            [--proxy proxy to use]
            [publisher]
```

Display additional information about the publisher:

```
root@solaris:~# pkg publisher sfe

            Publisher: sfe
                Alias:
           Origin URI: http://pkg.openindiana.org/sfe/
              SSL Key: None
             SSL Cert: None
          Client UUID: e873ddf6-3fba-11e2-8b21-880027ed6b01
      Catalog Updated: December  5, 2012 08:40:30 PM
              Enabled: Yes
```

Now remove the **OpenIndiana** publisher:

```
root@solaris:~# pkg unset-publisher sfe
root@solaris:~# pkg publisher
PUBLISHER                   TYPE     STATUS P LOCATION
solaris                     origin   online F http://pkg.oracle.com/solaris/release/
```

List all available packages on the system:

```
root@solaris:~# pkg list | more
NAME (PUBLISHER)                                  VERSION                    IFO
archiver/gnu-tar                                  1.26-0.175.1.0.0.24.0      i--
```

```
audio/audio-utilities                          0.5.11-0.175.1.0.0.24.2    i--
codec/flac                                     1.2.1-0.175.0.0.0.0.0      i--
codec/libtheora                                1.1.1-0.175.1.0.0.15.0     i--
codec/ogg-vorbis                               2.30.0-0.175.1.0.0.12.0    i--
codec/speex                                    1.2-0.175.1.0.0.15.0       i--
communication/im/pidgin                        2.10.5-0.175.1.0.0.24.0    i--
compress/bzip2                                 1.0.6-0.175.1.0.0.24.0     i--
compress/gzip                                  1.4-0.175.1.0.0.24.0       i--
compress/p7zip                                 9.20.1-0.175.1.0.0.24.0    i--
compress/unzip                                 6.0-0.175.1.0.0.24.0       i--
compress/xz                                    5.0.1-0.175.1.0.0.24.0     i--
compress/zip                                   3.0-0.175.1.0.0.24.0       i--
consolidation/SunVTS/SunVTS-incorporation      0.5.11-0.175.1.0.0.14.0    i--
consolidation/X/X-incorporation                0.5.11-0.175.1.0.0.24.1317 i--
consolidation/admin/admin-incorporation        0.5.11-0.175.1.0.0.5.0     i--
consolidation/cacao/cacao-incorporation        0.5.11-0.175.1.0.0.11.0    i--
consolidation/cde/cde-incorporation            0.5.11-0.175.0.0.0.0.0     i--
consolidation/cns/cns-incorporation            0.5.11-0.175.1.0.0.23.0    i--
consolidation/dbtg/dbtg-incorporation          0.5.11-0.175.1.0.0.15.0    i--
consolidation/desktop/desktop-incorporation    0.5.11-0.175.1.0.0.24.2    i--
consolidation/desktop/gnome-incorporation      0.5.11-0.175.1.0.0.22.0    i--
consolidation/gfx/gfx-incorporation            0.5.11-0.175.1.0.0.5.0     i--
consolidation/install/install-incorporation    0.5.11-0.175.1.0.0.24.1736 i--
consolidation/ips/ips-incorporation            0.5.11-0.175.1.0.0.24.0    i--
consolidation/java/java-incorporation          0.5.11-0.175.1.0.0.24.0    i--
consolidation/jdmk/jdmk-incorporation          0.5.11-0.173.0.0.0.0.0     i--
consolidation/l10n/l10n-incorporation          0.5.11-0.175.1.0.0.23.1134 i--
consolidation/man/man-incorporation            0.5.11-0.175.1.0.0.21.0    i--
consolidation/nspg/nspg-incorporation          0.5.11-0.175.1.0.0.5.0     i--
consolidation/nvidia/nvidia-incorporation      0.5.11-0.175.1.0.0.22.0    i--
consolidation/osnet/osnet-incorporation        0.5.11-0.175.1.0.0.24.2    i--
consolidation/sfw/sfw-incorporation            0.5.11-0.175.1.0.0.5.0     i--
consolidation/sic_team/sic_team-incorporation  0.5.11-0.175.1.0.0.5.0     i--
consolidation/solaris_re/solaris_re-incorporation 0.5.11-0.175.1.0.0.24.3  i--
```

```
consolidation/sunpro/sunpro-incorporation         0.5.11-0.175.1.0.0.19.0    i--
consolidation/ub_javavm/ub_javavm-incorporation   0.5.11-0.175.1.0.0.24.1    i--
consolidation/userland/userland-incorporation     0.5.11-0.175.1.0.0.24.0    i--
consolidation/vpanels/vpanels-incorporation       0.5.11-0.175.1.0.0.17.0    i-r
consolidation/xvm/xvm-incorporation               0.5.11-0.175.1.0.0.5.0     i--
crypto/ca-certificates                            0.5.11-0.175.1.0.0.24.2    i--
```

Note the IFO column:

Letter	Description
i	The package is installed
f	The package is *frozen* and cannot be updated
o	The package is obsolete

To understand the **r** in the third column of the **vpanels-incorporation** package, use the **info** subcommand:

```
root@solaris:~# pkg info vpanels-incorporation
          Name: consolidation/vpanels/vpanels-incorporation
       Summary:
         State: Installed (Renamed)
    Renamed to: pkg:/consolidation/userland/userland-incorporation
                pkg:/consolidation/osnet/osnet-incorporation
     Publisher: solaris
       Version: 0.5.11
 Build Release: 5.11
        Branch: 0.175.1.0.0.17.0
Packaging Date: May 29, 2012 10:02:23 PM
          Size: 5.46 kB
          FMRI: pkg://solaris/consolidation/vpanels/vpanels-
incorporation@0.5.11,5.11-0.175.1.0.0.17.0:20120529T220223Z
```

Use the **list** subcommand to easily identify the package version number of a single package:

```
root@solaris:~# pkg list vpanels-incorporation
```

```
NAME (PUBLISHER)                                VERSION                    IFO
consolidation/vpanels/vpanels-incorporation     0.5.11-0.175.1.0.0.17.0    i-r
```

To identify the package creation timestamp, use the **-v** switch of the **list** subcommand:

```
root@solaris:~# pkg list -v vpanels-incorporation
FMRI                                                                       IFO
pkg://solaris/consolidation/vpanels/vpanels-incorporation@0.5.11,5.11-0.175.1.0.0.17.0:20120529T220223Z i-r
```

The output of this command shows a line of the following format:

FMRI Scheme://Publisher/Package Name@Version,Build Release-Branch:Package Creation Timestamp IFO

The name **FMRI** means **Fault Managed Resource Identifier**. FMRIs are covered in the **Service Administration** lesson of this course.

Wildcards can be used to list packages available, either installed or not:

```
root@solaris:~# pkg list -a compress*
NAME (PUBLISHER)                                VERSION                    IFO
compress/bzip2                                  1.0.6-0.175.1.0.0.24.0     i--
compress/gzip                                   1.4-0.175.1.0.0.24.0       i--
compress/lcab (sfe)                             1.0.12-0.151.1.5           ---
compress/lcab/src (sfe)                         1.0.12-0.151.1.5           ---
compress/lha (sfe)                              1.14.9.1-0.151.1.5         ---
compress/lha/src (sfe)                          1.14.9.1-0.151.1.5         ---
compress/p7zip                                  9.20.1-0.175.1.0.0.24.0    i--
compress/unzip                                  6.0-0.175.1.0.0.24.0       i--
compress/unzoo (sfe)                            4.4-0.151.1.5              ---
compress/unzoo/src (sfe)                        4.4-0.151.1.5              ---
compress/xz                                     5.0.1-0.175.1.0.0.24.0     i--
compress/zip                                    3.0-0.175.1.0.0.24.0       i--
```

To display the number of installed packages, use the following command:

```
root@solaris:~# pkg list | wc -l
     873
```

To display the number of installed and available packages, use the **-a** switch:

```
root@solaris:~# pkg list -a | wc -l
    2615
```

Installing a Package

You are now going to install the ClamAV anti-virus package. This package is available from the **OpenIndiana** publisher. In order to use that publisher you need to add it once more:

```
root@solaris:~# pkg set-publisher -g http://pkg.openindiana.org/sfe sfe
root@solaris:~# pkg publisher
PUBLISHER                   TYPE     STATUS P LOCATION
solaris                     origin   online F http://pkg.oracle.com/solaris/release/
sfe                         origin   online F http://pkg.openindiana.org/sfe/
```

There are two ways to identify the package name before being able to install it, either by using the **list** subcommand:

```
root@solaris:~# pkg list -a | grep clamav
antivirus/clamav (sfe)                          0.97.6-0.151.1.6          ---
antivirus/clamav/src (sfe)                      0.97.6-0.151.1.6          ---
security/clamav (sfe)                           0.97.3-0.151.1            --r
```

or, alternatively by using the **search** subcommand:

```
root@solaris:~# pkg search antivirus/clamav
INDEX      ACTION VALUE                                     PACKAGE
```

```
require    depend pkg:/antivirus/clamav@0.97.5,5.11-0.151.1.5 pkg:/security/clamav@0.97.3-0.151.1
pkg.fmri   set    sfe/antivirus/clamav                        pkg:/antivirus/clamav@0.97.5-0.151.1.5
```

Installing ClamAV is now very simple:

```
root@solaris:~# pkg install antivirus/clamav
           Packages to install:  3
       Create boot environment: No
Create backup boot environment: No

DOWNLOAD                                  PKGS         FILES    XFER (MB)   SPEED
Completed                                  3/3         55/55      1.7/1.7  458k/s

PHASE                                          ITEMS
Installing new actions                       110/110
Updating package state database                 Done
Updating image state                            Done
Creating fast lookup database                   Done
```

To check whether the package is now installed, use the **list** subcommand:

```
root@solaris:~# pkg list clamav
NAME (PUBLISHER)                                VERSION                    IFO
antivirus/clamav (sfe)                          0.97.6-0.151.1.6           i--
```

or the **verify** subcommand:

```
root@solaris:~# pkg verify -v clamav
PACKAGE                                                                    STATUS
pkg://sfe/antivirus/clamav                                                    OK
```

Remember that for more information about the installed package, you can use the **info** subcommand:

```
root@solaris:~# pkg info clamav
```

```
          Name: antivirus/clamav
       Summary: Unix anti-virus scanner
      Category: Applications/System Utilities
         State: Installed
     Publisher: sfe
       Version: 0.97.6
 Build Release: 5.11
        Branch: 0.151.1.6
Packaging Date: October  1, 2012 08:13:32 PM
          Size: 3.95 MB
          FMRI: pkg://sfe/antivirus/clamav@0.97.6,5.11-0.151.1.6:20121001T201332Z
```

By using the **which** command, you will notice that there is no clamav program in **/bin** or **/usr/bin**:

```
root@solaris:~# which clamav
no clamav in /usr/bin /usr/sbin
```

So, either the program is not called clamav or it is elsewhere in the filesystem. In order to see which files the package installed, use the **contents** subcommand:

```
root@solaris:~# pkg contents antivirus/clamav
PATH
etc
etc/clamav-milter.conf
etc/clamd.conf
etc/freshclam.conf
usr
usr/bin
usr/bin/clamav-config
usr/bin/clambc
usr/bin/clamconf
usr/bin/clamdscan
usr/bin/clamdtop
usr/bin/clamscan
```

```
usr/bin/freshclam
usr/bin/sigtool
usr/include
usr/include/clamav.h
usr/lib
usr/lib/libclamav.so
usr/lib/libclamav.so.6
usr/lib/libclamav.so.6.1.15
usr/lib/libclamunrar.so
usr/lib/libclamunrar.so.6
usr/lib/libclamunrar.so.6.1.15
usr/lib/libclamunrar_iface.so
usr/lib/libclamunrar_iface.so.6
usr/lib/libclamunrar_iface.so.6.1.15
usr/lib/pkgconfig
usr/lib/pkgconfig/libclamav.pc
usr/sbin
usr/sbin/clamav-milter
usr/sbin/clamd
usr/share
usr/share/doc
usr/share/doc/clamav
usr/share/doc/clamav/COPYING
usr/share/doc/clamav/COPYING.LGPL
usr/share/doc/clamav/COPYING.bzip2
usr/share/doc/clamav/COPYING.file
usr/share/doc/clamav/COPYING.getopt
usr/share/doc/clamav/COPYING.llvm
usr/share/doc/clamav/COPYING.lzma
usr/share/doc/clamav/COPYING.regex
usr/share/doc/clamav/COPYING.sha256
usr/share/doc/clamav/COPYING.unrar
usr/share/doc/clamav/COPYING.zlib
usr/share/doc/clamav/ChangeLog
```

```
usr/share/doc/clamav/FAQ
usr/share/doc/clamav/README
usr/share/doc/clamav/clamav-mirror-howto.pdf
usr/share/doc/clamav/clamdoc.pdf
usr/share/doc/clamav/phishsigs_howto.pdf
usr/share/doc/clamav/signatures.pdf
usr/share/man
usr/share/man/man1
usr/share/man/man1/clambc.1
usr/share/man/man1/clamconf.1
usr/share/man/man1/clamdscan.1
usr/share/man/man1/clamdtop.1
usr/share/man/man1/clamscan.1
usr/share/man/man1/freshclam.1
usr/share/man/man1/sigtool.1
usr/share/man/man5
usr/share/man/man5/clamav-milter.conf.5
usr/share/man/man5/clamd.conf.5
usr/share/man/man5/freshclam.conf.5
usr/share/man/man8
usr/share/man/man8/clamav-milter.8
usr/share/man/man8/clamd.8
var
var/clamav
```

As you can see, the package installed the following programs:

```
usr/bin/clamav-config
usr/bin/clambc
usr/bin/clamconf
usr/bin/clamdscan
usr/bin/clamdtop
usr/bin/clamscan
usr/bin/freshclam
```

```
usr/bin/sigtool
usr/sbin/clamav-milter
usr/sbin/clamd
```

Uninstalling a Package

Uninstalling the package is now as simple as installing it:

```
root@solaris:~# pkg uninstall clamav
            Packages to remove:  3
       Create boot environment: No
Create backup boot environment: No

PHASE                                          ITEMS
Removing old actions                         101/101
Updating package state database                 Done
Updating package cache                           3/3
Updating image state                            Done
Creating fast lookup database                   Done
```

The **uninstall** subcommand removes all files installed by a package, **except** those that are dependencies of another installed package.

Searching for a Package containing a specific Program

You can also search for a package that contains a specific program:

```
root@solaris:~# pkg search /usr/bin/clamdscan
INDEX      ACTION VALUE             PACKAGE
```

```
path       file   usr/bin/clamdscan pkg:/antivirus/clamav@0.97.5-0.151.1.5
path       file   usr/bin/clamdscan pkg:/security/clamav@0.97.3-0.151.1
```

Updating Packages

The pkg command can also be used to update installed packages:

```
root@solaris:~# pkg update
No updates available for this image.
```

If you wish to update a specific package to the latest available version, use the following syntax : *pkg update package@latest*.

Solaris 11 Boot Environments

Solaris Boot Environments (BE) are an interaction between the **IPS** et **ZFS** technologies. Boot Environments offer an administrator a very efficient **rollback** facility. A Boot Environment is a ZFS file system that is designated for booting, in other words a bootable instance of the Operating System plus any other software added specifically to that image. It is possible to have as many Boot Environments as needed.

Imagine that you decide to update the Solaris kernel using IPS. In order to make sure that you can rollback to the previous version in the case of any problems, you would create a Boot Environment before updating. After updating the kernel you would have two Boot Environments, one before the update and one after the update.

Boot Environments are managed with the **beadm** command. As with the IPS pkg command, the Boot Environments' beadm command uses a set of subcommands.

beadm Subcommands

The beadm subcommands are as follows:

Subcommand	Description
list	Lists all available Boot Environments
list *BEname*	Lists the Boot Environment called *BEname*
create *BEname*	Creates a clone, called *BEname* of the **current** active Boot Environment. Options allow for creating from an inactive or snapshot Boot Environment
create *BEname*@*SNAPname*	Creates a snapshot called *SNAPname* of the Boot Environment *BEname*
activate *BEname*	Configures the Boot Environment *BEname* to be used at the next reboot
destroy *BEname*	Erases the Boot Environment *BEname* and all associated snapshots
mount *BEname*	Mounts the Boot Environment *BEname* on mount point for inspection
unmount *BEname*	Unmounts the Boot Environment *BEname*
rename *BEname newBEname*	Renames the Boot Environment *BEname* to *newBEname*

LAB #2 - Managing Boot Environments

Using the beadm Command

First, list the current available Boot Environments:

```
root@solaris:~# beadm list
BE               Active Mountpoint Space  Policy Created
--               ------ ---------- -----  ------ -------
solaris          NR     /          6.36G  static 2012-11-20 19:08
solaris-backup-1 -      -          159.0K static 2012-11-20 22:20
```

This output shows the **solaris** Boot Environment that was automatically created during the installation process together with the **solaris-backup-1** Boot Environment that was automatically created by the system after the installation process.

You will notice that the **Active** column shows two letters:

Letter	Description
N	The Boot Environment is active **N**ow
R	The Boot Environment will be active upon the next **R**eboot

Now create a new Boot Environment called **MyBE**:

```
root@solaris:~# beadm create MyBE
root@solaris:~# beadm activate MyBE
root@solaris:~# beadm list
BE               Active Mountpoint Space  Policy Created
--               ------ ---------- -----  ------ -------
MyBE             -      -          199.0K static 2012-12-07 10:43
solaris          NR     /          6.36G  static 2012-11-20 19:08
solaris-backup-1 -      -          159.0K static 2012-11-20 22:20
```

Note that the new Boot Environment **MyBE**, a clone of **solaris**, has been created but is not active. Clones are bootable images.

Activate the Boot Environment **MyBE** so it will used upon the next reboot:

```
root@solaris:~# beadm list
BE               Active Mountpoint Space  Policy Created
--               ------ ---------- -----  ------ -------
MyBE             R      -          6.36G  static 2012-12-07 10:43
solaris          N      /          45.0K  static 2012-11-20 19:08
solaris-backup-1 -      -          159.0K static 2012-11-20 22:20
```

Note here that the Boot Environment **MyBE** is now flagged to be used upon the next reboot.

At this point take a snapshot of the **solaris-backup-1** Boot Environment:

```
root@solaris:~# beadm create solaris-backup-1@SNAP-solaris-backup-1
root@solaris:/# beadm list -a solaris-backup-1
BE/Dataset/Snapshot                                      Active Mountpoint Space  Policy Created
-------------------                                      ------ ---------- -----  ------ -------
solaris-backup-1
   rpool/ROOT/solaris-backup-1                           -      -          113.0K static 2012-11-20 22:20
   rpool/ROOT/solaris-backup-1/var                       -      -          46.0K  static 2012-11-20 22:20
   rpool/ROOT/solaris-backup-1/var@SNAP-solaris-backup-1 -      -          0      static 2012-12-07 10:48
   rpool/ROOT/solaris-backup-1@SNAP-solaris-backup-1     -      -          0      static 2012-12-07 10:48
```

A **snapshot** is a read-only, non-bootable image of a Boot Environment at a given point in time.

As you can see the snapshot is created in the **ROOT dataset**. A dataset is a generic term for a ZFS filesystem, zone, snapshot or volume.

I will be telling you more about the ZFS filesystem in the **Storage Administration** lesson of this course.

If you take a look at the current ZFS configuration you will see the following:

```
root@solaris:/# zfs list
NAME                          USED  AVAIL  REFER  MOUNTPOINT
rpool                        6.99G  12.3G  4.58M  /rpool
rpool/ROOT                   4.84G  12.3G    31K  legacy
rpool/ROOT/MyBE              4.84G  12.3G  3.90G  /
rpool/ROOT/MyBE/var           863M  12.3G   205M  /var
rpool/ROOT/solaris             83K  12.3G  3.90G  /
rpool/ROOT/solaris-backup-1   113K  12.3G  1.98G  /
```

```
rpool/ROOT/solaris-backup-1/var    46K  12.3G   758M  /var
rpool/ROOT/solaris/var             83K  12.3G   205M  /var
rpool/VARSHARE                     57K  12.3G    57K  /var/share
rpool/dump                       1.03G  12.4G  1.00G  -
rpool/export                     77.4M  12.3G    32K  /export
rpool/export/home                77.4M  12.3G    32K  /export/home
rpool/export/home/trainee        77.4M  12.3G  77.4M  /export/home/trainee
rpool/swap                       1.03G  12.4G  1.00G  -
```

In the above output you can see that the ROOT dataset contains each Boot Environment's **Critical Dataset**. Any other **Shared Datasets**, such as **/export** which contain mount points common to both the active and inactive boot environments are located outside the root dataset area of each boot environment.

Use the **destroy** subcommand to erase the **MyBE** Boot Environment:

```
root@solaris:/# beadm destroy MyBE
Are you sure you want to destroy MyBE?  This action cannot be undone(y/[n]): y
The BE that was just destroyed was the 'active on boot' BE.
solaris is now the 'active on boot' BE. Use 'beadm activate' to change it.
```

Note that the solaris Boot Environment has now been activated.

Check the zfs configuration:

```
root@solaris:/# zfs list
NAME                          USED  AVAIL  REFER  MOUNTPOINT
rpool                        6.99G  12.3G  4.58M  /rpool
rpool/ROOT                   4.84G  12.3G    31K  legacy
```

```
rpool/ROOT/solaris               4.84G  12.3G  3.90G  /
rpool/ROOT/solaris-backup-1       113K  12.3G  1.98G  /
rpool/ROOT/solaris-backup-1/var    46K  12.3G   758M  /var
rpool/ROOT/solaris/var            863M  12.3G   205M  /var
rpool/VARSHARE                   57.5K  12.3G  57.5K  /var/share
rpool/dump                       1.03G  12.4G  1.00G  -
rpool/export                     77.6M  12.3G    32K  /export
rpool/export/home                77.5M  12.3G    32K  /export/home
rpool/export/home/trainee        77.5M  12.3G  77.5M  /export/home/trainee
rpool/swap                       1.03G  12.4G  1.00G  -
```

If you want to look at the contents of a specific Boot Environment, all you have to do is mount it:

```
root@solaris:/# mkdir /tmp/be
root@solaris:/# beadm mount solaris-backup-1 /tmp/be
root@solaris:/# ls -l /tmp/be
total 104
lrwxrwxrwx   1 root     root           9 Nov 20 19:19 bin -> ./usr/bin
drwxr-xr-x   6 root     sys            9 Nov 20 19:19 boot
drwxr-xr-x   2 root     root           4 Nov 20 19:09 cdrom
drwxr-xr-x  29 root     sys          239 Nov 20 19:25 dev
drwxr-xr-x   5 root     sys            5 Nov 20 19:25 devices
drwxr-xr-x  74 root     sys          165 Nov 20 21:40 etc
drwxr-xr-x   2 root     root           2 Nov 20 19:27 export
dr-xr-xr-x   2 root     root           2 Sep 19 22:09 home
drwxr-xr-x  19 root     sys           19 Nov 20 19:09 kernel
drwxr-xr-x  13 root     bin          301 Nov 20 19:09 lib
drwxr-xr-x   2 root     root           4 Nov 20 19:27 media
drwxr-xr-x   2 root     sys            2 Nov 20 19:19 mnt
dr-xr-xr-x   2 root     root           2 Nov 20 19:53 net
dr-xr-xr-x   2 root     root           2 Nov 20 19:53 nfs4
drwxr-xr-x   2 root     sys            2 Sep 19 22:09 opt
drwxr-xr-x   5 root     sys            5 Sep 19 22:09 platform
dr-xr-xr-x   2 root     root           2 Sep 19 22:09 proc
```

```
drwx------   2 root     root           4 Nov 20 19:19 root
drwxr-xr-x   2 root     root           2 Nov 20 19:08 rpool
lrwxrwxrwx   1 root     root          10 Nov 20 19:19 sbin -> ./usr/sbin
drwxr-xr-x   5 root     root           5 Sep 19 22:09 system
drwxrwxrwt   2 root     sys            2 Nov 20 19:25 tmp
drwxr-xr-x  30 root     sys           42 Nov 20 19:18 usr
drwxr-xr-x  34 root     sys           41 Nov 20 19:19 var
```

Finally, to update the Operating System Kernel, you can use the following command : **pkg update-image**. This command first makes a clone of the current Boot Environment, proceeds with downloading and installing all required packages into the new clone and finally activates the clone for startup on the next reboot.
